

KOMPRESI FILE.TXT DENGAN ALGORITMA HUFFMAN
PADA FILE DENGAN MENGGUNAKAN PENGKODEAN

BASE-64

SKRIPSI



Oleh :

LINGGA ADI FIRMANTO

(0734010127)

..

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS PEMBANGUNANNASIONAL "VETERAN" JATIM
SURABAYA
2012

Judul : KOMPRESI FILE.TXT DENGAN ALGORITMA HUFFMAN PADA FILE DENGAN
MENGGUNAKAN PENGODEAN BASE-64
Pembimbing I : Dian Puspita Hapsari, S.Kom, M.Kom
Pembimbing II : Achmad Junaidi, S.Kom.
Penyusun : Lingga Adi Firmanto

ABSTRAK

Kompresi data adalah upaya untuk memperkecil ukuran data tanpa kehilangan data aslinya. Sejumlah algoritma telah dikembangkan untuk memperoleh hasil kompresi sekecil mungkin, yang salah satunya adalah Huffman. Algoritma Huffman merupakan algoritma kompresi data berbasis statistik. Keunikan dari algoritma ini adalah semakin banyak terdapat karakter yang sama maka hasil kompresinya akan semakin kecil karena sejumlah karakter yang sama tersebut akan diwakili oleh satu pola saja. Penelitian ini berupaya agar data yang akan dikompresi memiliki “nilai kembar” sebanyak mungkin agar saat dikompresi dengan algoritma Huffman diperoleh hasil kompresi yang semakin kecil dengan melakukan pengkodean (encoding) berbasis 64 sebagai upaya yang bisa dilakukan untuk memperkecil interval. Terbatasnya jumlah karakter pola yang tersedia akan membuat peluang terjadinya karakter kembar semakin besar. Hasil akhir proses encode adalah berupa dokumen baru yang hanya terdiri dari susunan karakter pola saja. File ini selanjutnya akan dikompresi dengan algoritma Huffman.

Oleh karena itu, dalam tugas akhir ini, penulis membuat aplikasi yang mampu melakukan proses kompresi menggunakan algoritma huffman pada file yang mengalami pengkodean base-64 menggunakan bahasa pemrograman Java.

Kata kunci : kompresi, dekompresi, encoding base64, huffman, Java

xKATA PENGANTAR

Alhamdulillah rabbil ‘alamin, segala puji bagi Allah Yang Maha Kuasa yang telah memberikan kekuatan-Nya sehingga penulis bisa menyelesaikan tugas akhir yang berjudul : “Kompresi File Dengan Algoritma Huffman Pada File Dengan Menggunakan Pengkodean Base-64”. Serta kepada Nabi Muhammad SAW yang telah menuntun kita semua kejalan yang lurus dan di ridhoi oleh Allah SWT.

Melalui tugas akhir ini, penulis merasa mendapat kesempatan besar untuk lebih memperdalam ilmu pengetahuan yang diperoleh selama di perkuliahan. Terutama di bidang pemrograman java. Namun demikian, penulis menyadari bahwa Tugas Akhir ini masih memiliki banyak kelemahan dan kekurangan. Oleh karena itu penulis sangat mengharapkan kritik dan sarannya untuk pengembangan ke depannya.

Secara khusus, dalam kesempatan ini pula, penulis ingin mengucapkan terima kasih dan penghargaan sebesar-besarnya kepada:

1. Bapak Ir. Sutiyono MT selaku Dekan Fakultas Teknologi Industri Universitas Pembangunan Nasional “Veteran” Jawa Timur.
2. Ibu Dr. Ir. Ni Ketut Sari, MT selaku Ketua Program Studi Teknik Informatika Universitas Pembangunan Nasional ”Veteran” Jawa Timur.
3. Ibu Dian Puspita Hapsari, S.Kom, M.Kom, dan Bapak Achmad Junaidi, S.Kom selaku Dosen Pembimbing Teknik Informatika Universitas Pembangunan Nasional ”Veteran” Jawa Timur.

4. Kedua orang tua saya dan keluarga tercinta yang telah memberikan semangat dan do'a restunya yang tiada henti.
5. Teman-teman seperjuangan Indra,kiki,reza(ahong),farid,yursa,nanang yang selalu ada untuk membantu meringankan pengerjaan tugas akhir ini.
6. Rizky Novitasari atas dukungannya selama ini serta teman-teman dari drughi indonesia.

Saya menyadari bahwa dalam penyusunan laporan ini tidak terlepas dari kekurangan dan kesalahan. Untuk itu, saya sangat terbuka bagi kritik dan saran yang bersifat membangun. Semoga laporan tugas akhir ini bermanfaat bagi saya khususnya dan bagi pembaca umumnya.

Surabaya, 15 Juni 2012

Penulis

DAFTAR ISI

ABSTRAK.....	i
KATA PENGANTAR	ii
DAFTAR ISI	iv
DAFTAR GAMBAR.....	vii
DAFTAR TABEL.....	viii

BAB I PENDAHULUAN

1.1 Latar Belakang Masalah	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah.....	3
1.4 Tujuan.....	4
1.5 Manfaat	4
1.6 Metodologi Penelitian.....	4
1.7 Sistematika penulisan	6

BAB II TINJAUAN PUSTAKA

2.1 Representasi Data	7
2.1.1 BCD (Binary Coded Decimal)	7
2.1.2 SBCDIC (Standard Binary Coded Decimal).....	8
2.1.3 EBCDIC (Extended Binary Coded Decimal).....	8
2.1.4 ASCII (American Standard Code for Information Interchange).....	8
2.2 Konsep Dasar Informasi	9
2.3 Kompresi Data	11
2.4 Metode Kompresi Data.....	13
2.5 Teknik Kompresi Data.....	14
2.6 Algoritma Huffman	15
2.7 Pengkodean Data.....	18
2.8 Algoritma Pengkodean Base64.....	19
2.8.1 Decoding Base 64 menjadi ASCII.....	20

2.9 Unified Modelling Language	21
2.10 Semantik Dalam UML.....	26
2.11 Notasi dalam UML	28
2.11.1 Actor.....	28
2.11.2 Use Case.....	29
2.11.3 Class Diagram.....	30
2.11.4 Interface.....	30
2.11.5 Interaction.....	31
2.11.6 Note.....	31
2.11.7 Dependency	31
2.11.8 Association	32
2.11.9 Generalization.....	32
2.11.10 Realization	33
2.12 Bahasa Pemrograman Java	33
BAB III Analisis Kebutuhan Dan Perancangan Sistem	
3.1 Analisa Kebutuhan	40
3.2 Pemodelan Fungsional.....	40
3.2.1 Diagram Alir.....	41
3.2.2 Kelas Diagram	42
3.2.3 Pseudocode Encoding Base64.....	44
3.2.4 Pseudocode Decoding Base64.....	45
3.2.5 Pseudocode Kompresi Huffman.....	46
3.2.6 Pseudocode Dekompresi Huffman	46
3.3 Use Case Diagram	47
3.4 Sequence Diagram.....	48
3.4.1 Sequence Diagram Compress.....	48
3.4.2 Sequence Diagram Decompress	49
3.5 Perancangan Sistem.....	50
3.6 Perancangan Antar Muka.....	50
3.6.1 Perancangan Form Untuk Encoding Dan Kompresi	51
3.6.2 Perancangan Form Untuk Decoding Dan Dekompresi	51

BAB IV IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi Hasil Perancangan	53
4.2 Implementasi Package Base64.....	53
4.2.1 Implementasi Fungsi byte[] Encode ()	53
4.2.2 Implementasi Fungsi int Encode().....	54
4.2.3 Implementasi Fungsi byte[] Decode()	54
4.2.4 Implementasi Fungsi int Decode()	55
4.3 Implementasi Package Frequency Table	55
4.3.1 Implementasi Fungsi int getSymbolLimit()	55
4.3.2 Implementasi Fungsi int Get().....	56
4.3.3 Implementasi Fungsi Set()	56
4.3.4 Implementasi Fungsi increment ().....	56
4.4 Implementasi Package Huffman Compress.....	57
4.4.1 Implementasi Fungsi FrequencyTable getFrequencies()	57
4.4.2 Implementasi Fungsi WriteCode()	58
4.4.3 Implementasi Fungsi Compress()	58
4.5 Pengujian Sistem	59
4.5.1 Pengujian Encoding dan Kompresi	59
4.5.2 Pengujian Decoding dan Dekompresi.....	63

BAB V PENUTUP

5.1 Kesimpulan.....	67
5.2 Saran.....	67

DAFTAR PUSTAKA

D A F T A R G A M B A R

Gambar 2.1	Siklus Informasi.....	10
Gambar 2.2	Komponen Kualitas Informasi	11
Gambar 2.3	Representasi teks ABACCDA.....	16
Gambar 2.4	Pohon Huffman untuk teks ABACCDA.....	17
Gambar 2.5	Metodologi dalam UML	23
Gambar 2.6	Notasi Actor	28
Gambar 2.7	Notasi Use Case.....	29
Gambar 2.8	Notasi Class	30
Gambar 2.9	Notasi Interface	30
Gambar 2.10	Notasi Interaction	31
Gambar 2.11	Notasi Note.....	31
Gambar 2.12	Notasi Dependency	31
Gambar 2.13	Notasi Association	32
Gambar 2.14	Notasi Generalization.....	32
Gambar 2.15	Notasi Realization.....	33
Gambar 3.1	Diagram Alir Aplikasi encoding dan kompresi.....	41
Gambar 3.2	Kelas Diagram Aplikasi Encode64 dan Kompresi Huffman	42
Gambar 3.3	Use Case Diagram Aplikasi Kompresi Data	47
Gambar 3.4	Sequence Diagram Compress.....	48
Gambar 3.5	Sequence Diagram Decompress	49
Gambar 3.6	Rancangan Form Kompresi	51
Gambar 3.7	Rancangan Form Dekompresi.....	52
Gambar 4.1	Pengujian encoding dan kompresi	59
Gambar 4.2	Grafik Hasil Uji Coba Kompresi	62
Gambar 4.3	Pengujian decoding dan dekompresi	63
Gambar 4.4	Grafik Hasil Uji Coba Dekompresi	65

DAFTAR TABEL

Tabel 2.1	Kode Huffman untuk “ABACCD A”	17
Tabel 2.2	Contoh Encoding dengan Metode Base-64.....	20
Tabel 2.3	Perbandingan Java dengan Bahasa Lain	35
Tabel 4.1	Rangkuman Hasil Percobaan Encoding dan Kompresi	60
Tabel 4.2	Rangkuman Hasil Percobaan Decoding dan Decompress	63

BAB I

PENDAHULUAN

1.1. Latar Belakang

Kompresi data adalah upaya untuk memperkecil ukuran data tanpa kehilangan data aslinya. Bisa dikatakan bahwa hasil akhir dari proses kompresi data adalah terbentuknya data baru yang merupakan modifikasi dari data asli dengan algoritma tertentu dimana data baru ini memiliki ukuran lebih kecil daripada data aslinya, Data hasil kompresi tidak bisa dibaca secara langsung sebagaimana data asli; data hasil kompresi bisa dibaca setelah sebelumnya di-dekompresi terlebih dahulu.

Sejumlah algoritma telah dikembangkan untuk memperoleh hasil kompresi sekecil mungkin. Kebanyakan dari algoritma tersebut merupakan pengembangan maupun perbaikan dari algoritma yang sudah ada, salah satunya dikembangkan dari algoritma Huffman. Algoritma Huffman merupakan algoritma kompresi data berbasis statistik. Struktur tree untuk menyusun karakter kompresi diperoleh melalui perhitungan statistik berdasarkan jumlah kemunculan tiap karakter. Keunikan dari algoritma ini adalah semakin banyak terdapat karakter yang sama maka hasil kompresinya akan semakin kecil karena sejumlah karakter yang sama tersebut akan diwakili oleh satu pola saja. Penelitian ini berupaya agar data yang akan dikompresi memiliki “nilai kembar” sebanyak mungkin agar saat dikompresi dengan algoritma Huffman diperoleh hasil kompresi yang semakin kecil.

Peluang untuk memperoleh data kembar ditentukan oleh panjang-pendeknya interval kode karakter yang digunakan. Secara umum karakter di komputer dikodifikasi dalam format ASCII (American Standard Code for Information Interchange) dengan panjang interval 256, yaitu kode 0 sampai 255. Tiap karakter akan disimpan oleh komputer dalam bentuk biner sebanyak 8 bit. Dalam format ASCII, setiap karakter akan memiliki nilai biner antara 00000000 sampai dengan 11111111. Dengan panjang interval 256 maka tiap karakter akan memiliki formasi biner berbeda dibandingkan dengan karakter lainnya. Bila panjang interval ini diperkecil maka akan ada sejumlah karakter berbeda yang memiliki nilai biner sama. Upaya yang bisa dilakukan untuk memperkecil interval ini adalah dengan proses encode data.

Encode merupakan upaya untuk mengubah suatu karakter menjadi karakter lain yang sudah disediakan. Ini berbeda dengan enkripsi yang menggunakan persamaan matematika untuk mengubah suatu karakter menjadi karakter lain. Pada encode, karakter yang akan menjadi karakter pengganti sudah tersedia, tinggal harus dihitung berdasarkan rumus tertentu untuk mencari karakter mana dalam pola yang harus digunakan untuk menggantikan karakter tertentu.

Algoritma Base-64 merupakan salah satu algoritma untuk proses encode data dengan menyediakan karakter pola sebanyak 64 buah, yaitu ABCDEFGHIJKLMNOPQRSTUVWXYZ, abcdefghijklmnopqrstuvwxyz, 0123456789, +/-. Karakter pola tersebut hanya menggunakan 6 bit data. Karakter yang di-encode dengan algoritma base 64 adalah karakter dengan lebar 8 bit, diubah menjadi karakter pola dengan lebar 6 bit. Terbatasnya jumlah karakter pola yang tersedia akan membuat peluang terjadinya karakter kembar semakin besar.

Hasil akhir proses encode adalah berupa dokumen baru yang hanya terdiri dari susunan karakter pola saja. File ini selanjutnya akan dikompresi dengan algoritma Huffman.

Oleh karena itu, dalam tugas akhir ini, penulis akan membuat aplikasi yang mampu melakukan proses kompresi menggunakan algoritma huffman pada file yang mengalami pengkodean base-64.

1.2. Rumusan Masalah

Berdasarkan latar belakang masalah di atas, permasalahan dalam penelitian ini adalah : Bagaimana membuat aplikasi yang dapat menggabungkan konsep encode base 64 dengan algoritma Huffman ?

1.3. Batasan Masalah

Agar tidak menyimpang dari tujuan yang ingin dicapai, penelitian ini dibatasi pada hal-hal sebagai berikut:

1. Karakter yang ditulis dalam naskah adalah meliputi karakter-karakter yang terdapat dalam tabel ASCII.
2. Penelitian ini dilaksanakan pada file berekstensi *.txt.
3. Penelitian ini tidak melakukan perbandingan antara hasil kompresi file asli dengan hasil kompresi file yang telah di-encode sebelumnya.
4. Penelitian ini tidak membahas tentang banyaknya waktu yang dibutuhkan untuk melakukan proses kompresi / dekompresi. Istilah kompresi di sini sudah meliputi kegiatan encode sebagai pendahuluan.

5. Aplikasi ini dapat berjalan pada sistem operasi Microsoft Windows XP, Vista dan Seven

1.4. Tujuan

Adapun tujuan dari penelitian ini adalah menghasilkan aplikasi yang dapat menggabungkan konsep encode data dengan kompresi data serta untuk pengamanan file.txt setelah proses encoding.

1.5. Manfaat

Adapun manfaat yang ingin diperoleh dari pengerjaan tugas akhir ini adalah dapat membuat perangkat lunak untuk mempermudah pengguna dalam proses pengkodean base-64 sekaligus melakukan proses kompresi menggunakan algoritma huffman.

1.6. Metodologi Penelitian

Metode penelitian merupakan tahapan-tahapan yang dilalui oleh peneliti dari perumusan masalah sampai kesimpulan, yang membentuk sebuah alur yang sistematis. Metodologi penelitian ini digunakan sebagai pedoman penelitian dalam pelaksanaan penelitian ini agar hasil yang dicapai tidak menyimpang dari tujuan yang telah ditetapkan sebelumnya. Adapun metode penelitian yang dipergunakan dalam pengerjaan tugas akhir ini adalah :

a. Studi Literatur

Mencari referensi dan bahan pustaka tentang teori-teori yang berhubungan dengan permasalahan yang akan dikerjakan dalam tugas akhir ini.

b. Studi Kasus

Mencari contoh-contoh kasus serupa yang berhubungan dengan permasalahan dalam tugas akhir ini.

c. Analisis dan Perancangan

Membuat analisa berdasarkan data-data yang sudah dimiliki, membuat model matematisnya dan merancang alur penyelesaian berdasarkan algoritma Huffman dan encoding base64. Perancangan aplikasi dimulai dengan perancangan antar muka aplikasi, kemudian merancang detail kombinasi dari kedua algoritma tersebut.

d. Implementasi Program

Mengimplementasikan teknik algoritma yang akan digunakan. Detail mengenai implementasi program dilakukan sesuai hasil analisis dan perancangan aplikasi pada tahapan sebelumnya.

e. Pengujian Aplikasi

Pengujian dilakukan pada aplikasi yang telah dibuat. Menguji validitas dan efektifitas algoritma yang diterapkan pada aplikasi.

f. Evaluasi dan Penarikan kesimpulan

Evaluasi dilakukan untuk mengetahui kinerja aplikasi kompresi data teks sesuai ukuran dan format data teksnya, selanjutnya dilakukan penarikan kesimpulan.

1.7. Sistematika Penulisan

Sistematika penulisan tugas akhir ini disusun untuk memberikan gambaran umum tentang penelitian yang dijalankan. Sistematika penulisan tugas akhir ini adalah sebagai berikut :

BAB I PENDAHULUAN

Bab ini berisi latar belakang masalah, identifikasi masalah, maksud dan tujuan yang ingin dicapai, batasan masalah, metodologi penelitian yang diterapkan dalam memperoleh dan mengumpulkan data, waktu dan tempat penelitian, serta sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Membahas berbagai konsep dasar dan teori-teori yang berkaitan dengan topik masalah yang diambil dan hal-hal yang berguna dalam proses analisis permasalahan.

BAB III ANALISIS DAN PERANCANGAN

Menganalisis masalah dari model penelitian untuk memperlihatkan keterkaitan antar variabel yang diteliti serta model matematis untuk analisisnya.

BAB IV IMPLEMENTASI DAN PENGUJIAN

Membahas mengenai pengimplementasian aplikasi yang telah dibuat ke perangkat yang akan digunakan serta melakukan pengujian terhadap aplikasi yang telah diimplementasikan tersebut.

BAB V PENUTUP

Berisi kesimpulan dan saran yang sudah diperoleh dari hasil penulisan tugas akhir.